

Documento de reflexión

Desde la ejecución del proyecto uno hasta la última entrega del proyecto tres, nosotros como grupo hemos experimentado diversos conflictos que nos han hecho de una manera u otra escalar en el proceso de conocimiento acerca de esta nueva tecnología que usamos y todo lo que lleva por detrás. A lo largo de este documento, analizaremos en detalle el desarrollo de los proyectos, evaluando lo que salió bien, lo que salió mal, las decisiones acertadas y problemáticas, así como los problemas enfrentados durante el desarrollo.

Proyecto 1

- El proyecto es un buen inicio para poner en evidencia la realidad del proceso para solucionar algún problema relacionado a programación antes de la implementación, la parte de creación del diseño tiende a ser un poco complicado, esto debido a que se estudian temas nuevos, como la programación orientada a objetos, un paradigma de programación que parte del concepto de "objetos" como base, los cuales contienen información en forma de campos y código en forma de métodos. La teoría puede ser un poco abrumadora, ya que, este tipo de estructura maneja diversos tópicos, sin embargo, a medida que uno iba haciendo uso de estos nuevos conceptos se iba familiarizando más y más hasta el punto de entender cómo funcionan estos y como se relacionan entre sí
- Consideramos la creación del metamodelo de clases, una de las partes más críticas para la realización de este proyecto. La creación de clases, las relaciones entre estas y la organización en general de la estructura del proyecto es muy importante, esto define la manera en que se va a organizar hasta el final el proyecto, las relaciones que van a existir entre las diversas clases que posee y su implementación. Acorde a nuestra experiencia pudimos evidenciar que no siempre se llega a un resultado correcto, es realmente un proceso evolutivo que a medida que se avanza en el proyecto se evidencian posibles revisiones y mejoras en este.
- El identificar los requerimientos funcionales también es de gran importancia, puede que no sea tan complejo el identificarlos, lo realmente importante es cubrir todos aquellos que se esperan. Un acierto es la creación del documento de análisis incluyendo los casos de uso y una tabla con todos los requerimientos funcionales y que es lo que se esperan que realicen.
- La implementación puede ser quizás uno de los puntos más complicados si aún no se está familiarizado con Java y el IDE eclipse, por lo general la estructura de programación se mantiene relativamente similar con respecto a Python, sin embargo, las cosas empiezan a cambiar cuando empezamos a aplicar la programación orientada a objetos. Un gran acierto es tener un buen documento de diseño, ya que, este lo guía a uno muy bien acerca de que hacer, que no hacer y como relacionar diferentes componentes del diseño.

- Un problema que tuvimos en la implementación fue el manejo de datos, se crearon los respectivos datos para la obtención de datos, sin embargo, nuestro proyecto no presentaba una adecuada persistencia de estos. Al final, entendimos que la mejor manera de manejar este tipo de problema es con métodos que permitan actualización de los archivos por medio de copias, es decir, métodos que copien la información de un archivo y cree uno nuevo con la información del anterior junto al ajuste actualizado.

Proyecto 2

- El boceto es de gran ayuda para tener una amplia idea de los componentes que nuestro proyecto iba a tener, además, se podía manejar los componentes en relación con los requerimientos funcionales necesarios, es decir, se podía tener idea de que paneles y componentes en general se necesitaban para un requerimiento en específico, a la hora de implementar iba a quitar la preocupación por cubrir estos requerimientos.
- Para el proyecto dos era muy importante tener conocimiento acerca del framework que se iba a manejar (SWING), un gran acierto fue mirar la documentación directamente desde la pagina de Oracle para tener conocimiento sobre métodos y relaciones de los componentes que maneja esta tecnología. Además, el realizar el taller 4 fue de gran ayuda, ya que, nos familiariza con el manejo de swing.
- Manejar el proyecto bajo el patrón MVC fue un gran acierto, dado el como se relacionaba la vista con un objeto controlador y este con clases modelo permitió una manera mas eficiente de manejar la interacción del sistema, además, de un mejor orden y entendimiento acerca del sistema.
- Una problemática que tuvimos fue dificultades en la integración de componentes Swing personalizados debido a la falta de experiencia. Esto al principio fue tedioso dada la poca experiencia que se tenía, sin embargo, con forme se iba realizando paneles y usando sus componentes se empezó a tener un manejo mas eficiente y con menores errores.
- El nuevo diseño es de suma importancia, este al igual que el interior ayuda de manera significativamente sobre la relación de los componentes del proyecto y como estos cubren los requerimientos funcionales solicitados. En general, es un proyecto no tan complejo, solo es importante tener un buen acierto acerca de los componentes que se van a usar y como estos se relacionan
- Los action listeners fueron métodos llamativos y entenderlos fue importante. Nosotros como grupo entendimos que el action listener no realiza cambios en la vista, si no, en el controlador y modelos que al final actualiza lo que el usuario observa según si interacción con el programa.

Proyecto 3

- El proyecto tres fue el mas amigable, se cree que es porque ya se tenia una previa experiencia de los dos anteriores proyectos. Un acierto para este proyecto fue tener las bases del proyecto 1 y proyecto 2 bien realizadas, ya que, nos permitió agregar código de manera acertada con una tasa de errores mínimos y que mantienen la funcionalidad del programa.
- El punto de generar un PDF no fue realmente desafiante, sin embargo, tuvimos problemas con respecto a la implementación de librerías externas, esto fue más una problemática de conocimiento acerca del build path del proyecto y de como este interactuaba en este. La solución fue simplemente informarse en diferentes fuentes sobre la implementación de librerías externas para nuestro uso. Después de esto, los métodos para generar los PDF salieron bastante fáciles.
- La implementación de las clases dinámicas fue quizás el mayor desafío del proyecto, ya que, no se tenia conocimiento previo sobre esta tecnología y además, por la implementación de los pagos se tuvo que cambiar bastante las clases ya realizada, esto se pudo prevenir haciendo uso del principio open/closed previendo la integración de nuevos atributos para el usuario. La implementación de clases dinámicas si fue necesario el realizar una investigación extensa para entender la manera de uso y varias pruebas de error para finalmente llegar al resultado esperado.
- La creación de una nueva interfaz para el cliente no denoto gran dificultad, esto debido a que, primero se tenía un buen modelo MVC que permitía acceder a los requerimientos del proyecto de una manera fácil y segundo por la previa experiencia del proyecto dos. Las pruebas tampoco fueron de gran dificultad y estas se pudieron realizar de manera óptima, al igual que la implementación de nuevos vehículos, ya que, solo se tenia que adaptar estos nuevos requerimientos como atributos, algo con lo que ya estábamos bastante familiarizados.

Conclusiones generales

El proyecto 1 que abarcan el análisis y diseño en Java han proporcionado una valiosa perspectiva sobre la importancia de una planificación cuidadosa, la toma de decisiones fundamentada y la adaptabilidad en el desarrollo de programas. A través de estas experiencias, hemos identificado tanto áreas de éxito como desafíos, ofreciendo lecciones clave para futuros proyectos similares. El proyecto 2 usando Java Swing proporcionó una valiosa experiencia en el desarrollo de interfaces gráficas, sin embargo, enfrentamos desafíos al lidiar con aspectos específicos, como la sincronización de eventos y el rendimiento. El proyecto 3 que reforzó el uso de Java Swing y la implementación de pruebas unitarias ha demostrado una mejora en términos de interfaces gráficas eficientes. Aunque enfrentamos desafíos, las lecciones aprendidas resaltan la importancia de una planificación cuidadosa, un enfoque equilibrado y adecuado en las pruebas, este proyecto sirve para mostrar el producto de buenas prácticas de desarrollo. En general, los tres proyectos son muy enriquecedores intelectual y prácticamente hablando, que sienta excelentes bases para la continuidad de creación de software mas complejos en el futuro.